

A Web Site Editing System based on Unfold/Fold Operations

Hironobu Aoki Motoki Miura Jiro Tanaka
Institute of Information Sciences and Electronics, University of Tsukuba
1-1-1 Tennodai, Tsukuba, Ibaraki, 305-8573, Japan

Abstract *A better page structure for a web site improves the usability of its contents. Though ensuring a well-designed structure is important, site updating, such as adding new contents, may contort the consistency of a site. Most editors have experienced trouble changing the site-structure because the tasks include both creating new pages and linking to the pages.*

We have designed and implemented a system called “Javara” for restructuring a web site. Javara is a kind of editor, but the principal operations are link unfolding and folding. Link unfolding inserts the linked document into the page, which can then be used to gather the sparse contents into a page. Folding means to create both a new page and a link to that page at the same time, which can be used to divide the contents of a grand page into sub pages. Editors can specify a target area to be exported anywhere with casual drag operations. Javara also has a zoomable pane to show a longer page. These facilities will reduce the editor’s burden.

Keywords: Web, Authoring Tool, HTML, DOM, Zooming

1 Introduction

1.1 Building a web site

A better page structure for a web site improves the usability of its contents. The common procedure for building a web site consists of three phases: plotting, editing and checking. During the plotting phase, designers consider the content hierarchy and divide the contents into individual pages and link structures. After the plotting phase, editors start authoring actual HTML files for each page in the editing phase. In the checking phase, editors browse the site to verify the display and the link structure.

1.2 Restructuring a web site

The web is dynamic, but repetitive changeovers and additions tend to distort a web site structure. For example, if the content is spread all over the site, readers have difficulty understanding the web site. Spelling errors or description errors on a single page can be corrected by conventional editing of that page. However, when it comes to an inappropriate site structure, editors are required to revise the structure of the whole site. One way to fix the structure problem is to break the page up into smaller pages. The other way is to merge the

content of the linked page into the base page. In these ways, authors must take care of both the content and the structure.

According to [1], most web site developers restructure web sites they have made previously. There is much research to support designers in finding a suitable structure by analyzing user’s browsing logs. However, there is little research to support editors in fixing the structure of the site. Even Microsoft FrontPage [2] and Macromedia Dreamweaver [3] have features to view the structure but not to correct it. Editors desiring to change the whole structure must frequently edit across the files.

In this paper, we introduce a new method for restructuring a web site. We named it the “Javara method.” “Javara” means “cornice” in Japanese. The Javara method helps editors to handle the structure and the content of the web site at the same time. We also developed an editing system based on this method.

2 Concept of Javara methods

In restructuring, it may useful to move the content across the base page and linked page because the linked document is designed to be read as an option. It may also be useful to divide a part of the page into a new page and put a link from the base to the new page.

These operations are similar in concept to controlling the amount of content on a page because a page may include a supplemental description that can be replaced in another page. A link-oriented approach may be useful for restructuring a site according to its context.

To make these operations natural, we propose applying the concept of “inlineLink”[4] to restructuring. “inlineLink” is a link following technique for browsing structured documents. It inserts the content of the linked page near the link anchor. Our Javara methods consist of two operations, “unfolding link” and “folding” (see Figure1). Unfolding and folding operations support our link-oriented restructuring approach.

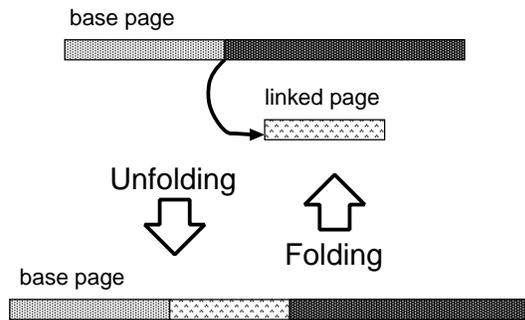


Figure 1: Unfolding and Folding

Unfolding a link moves the content of the linked page into the base page and removes the linked page from the site repository. The link anchor is replaced with the content of the linked page. The inserted region can be edited in the same way as the other part of the base page.

Folding exports selected contents of the base page to a new blank page. The selected region then becomes a link to the exported page. This operation is the inverse of unfolding. Editors can specify any part in the base page as a selected region.

inlineLink changes only the representation. In contrast, Javara can change the content and structure of the web site. Typical restructuring steps are: (1) Unfolding links to bundle small pieces of contents, (2) Selecting the region to be folded, and (3) Folding the selected part. The unfolded contents can be treated in the same way as the original base document, so there is no boundary to consider in specifying a region.

Restructuring with traditional methods is bounded by the file partition so that editors cannot edit the web site content intuitively. When editors want to copy or move contents across the file with conventional tools, they have to frequently switch target sources. Using our Javara method, editors can treat the entire site content in one window, so they do not need to switch target sources.

3 Design of Javara system

In this section, we discuss the system design and issues related to our methods.

3.1 Unfolding issues

If the system unfolds all kinds of links, the site structure may become inconsistent due to redundant text or looped links. To avoid this, the system must consider the link structure of the target pages.

Assume that the system tries to unfold a link to a page that has incoming links. When the linked page is deleted, the incoming links from other pages are broken. To avoid this, the system retains the linked page. The system warns the editors and confirms the redundancy of the copied context.

When the system unfolds a link, the linked page may include links to the base page. A typical one is the “Back” anchor. If the editor unfolds the back anchor, the base page becomes redundant and contains duplicated descriptions. A similar issue is the existence of a link that points to an anchor in the same page. In such cases, the system rejects the unfolding request with an alert.

3.2 Folding issues

Issues in folding are derived from tag mismatching. Editors can specify folding any part of the site by dragging. Though an HTML source snippet of the specified region can be retrieved through a browser, the snippets cannot be ensured to correspond to the HTML tags. To validate the snippet, the system adds extra HTML tags to both the base page and the new page to maintain the correspondence.

We now consider the case shown in Figure 2. The editor wants to fold the highlighted region. The HTML source of the highlighted region is shown below.

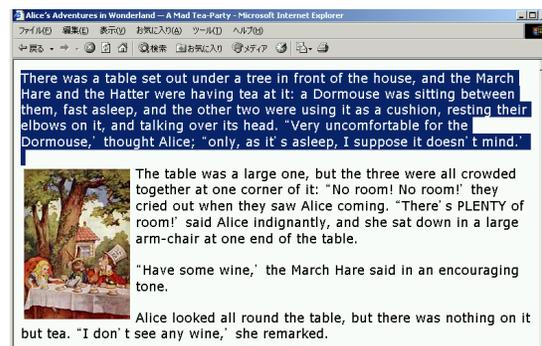


Figure 2: A specified region

```
<FONT face="palatino, verdana, times">
There was a table set out under a tree
in front of the house, and the March
Hare and the Hatter were having tea at
it: a Dormouse was sitting between them,
fast asleep, and the other two were using
it as a cushion, resting their elbows on it,
and talking over its head. "Very uncomfortable
for the Dormouse," thought Alice;" only,
as it's asleep, I suppose it doesn't mind.'
<BR><BR>
```

If the system simply folds this region, a problem occurs. There is a FONT tag at the head of this source. However, the tag to close this is not in this source but at the bottom of the source of this page. To create a valid HTML page, simple dividing is not enough. The base page may create a problem because the FONT tag is moved into a new page, so there is only a closing tag and the display differs from the original one.

In this sample, the system inserts `` at the top of the new page, and inserts `` after the folded region in the base page.

The system uses a template of HTML tags on the head and bottom to make a blank page. The template includes DOCTYPE declaration and HTML, HEAD, TITLE and BODY tags.

A region may be freely selected, the borders of a region sometimes divide existing anchors. Obviously, most such cases are just mistakes or operational slips, and the operation is not so meaningful because both the base page and the exporting page have the same link. Therefore, the system confirms with a yes/no dialog whether the editor really wants to fold at that point. If the editor forces the folding and divides anchors, the system writes the entire anchor label in both the base page and the exported page.

3.3 Mapping unfolding/folding operations

Editors click a link anchor to request unfolding. Figure 3 (a) shows a page before unfolding. When editors unfold a link, the content of the target page is inserted into the base page as in Figure 3 (b). The inserted contents can be edited in the same way as other contents.

Editors specify a region from the base page by mouse. When the editor hits the 'f' key on the keyboard, the region is folded. Editors specify the region to fold by dragging the text field (see Figure 4 (a)). Hitting the 'f' key folds the region. A text input window (Figure 4 (b)) appears to specify the label of the link anchor. Figure 4 (c) shows that the selected region is replaced with the link anchor.

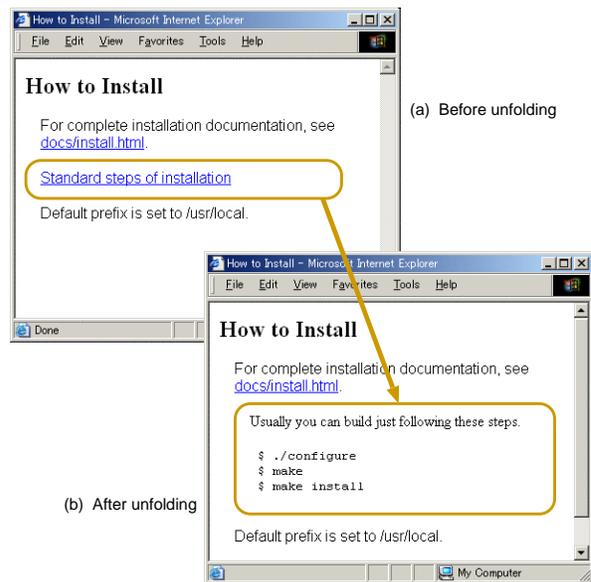


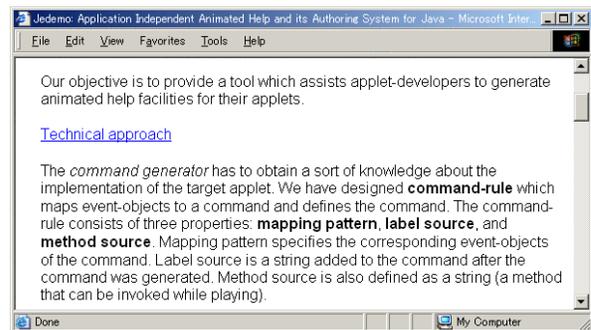
Figure 3: Unfolding operation



(a) Before Folding



(b) Text Input Window



(c) After Folding

Figure 4: Folding operation

3.4 Zooming display

3.4.1 Zooming out the whole page

It is important for editors to recognize the content of the whole site while editing by unfolding and folding. In particular, when they restructure the site, they must cut and paste or change the structure so that they often lose their mental map of the site.

Using Javara methods, editors can easily understand the relationship between the base page and linked page. However, unfolding makes the base page longer, and a longer page causes difficulty in editing the site. If the unfolded page is huge, the text after the anchor moves far from the original position. If many links are unfolded, the base page itself becomes very huge and the whole site cannot be displayed in a single window, making it difficult for editors to view the whole site.

We have applied a zooming technique to solve the above problems. The whole site can be zoomed to fit the window size, so that editors can view the content of the entire site. Each unfolding or folding changes the size of the full page and causes a change of zooming rate to fit the window size.

Though zooming out may help editors to recognize the contents, it may cause difficulty in editing because the display of the site becomes too small to read. To solve this problem, we used context zooming of a focused part. The system estimates a target part that an editor is trying to edit. The estimated area can be displayed larger than the other areas.

3.4.2 Zooming operation

The whole web site is displayed by zooming. When the length of the site is changed by unfolding or folding, the system adjusts the zoom rate to display the whole site in the window.

If the site becomes long and the text becomes too small to read due to zooming out, zooming into the focused part enables users to read and edit the text. Figure 5 shows an example of adjusting the zoom rate of both the whole site and the focused part. Editors can more easily edit the focused part than the other parts. The other parts are adjusted to fit the window, so that editors can also take in the entire site's content. When the focused part changes, the zoom rate also changes.

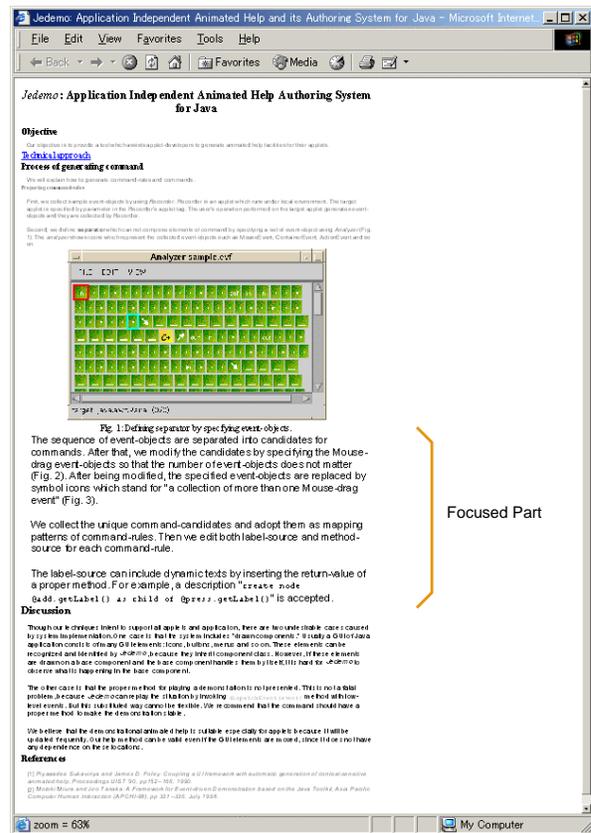


Figure 5: Zooming in the focused part; the editor can read the text of the focused part.

4 Implementation

4.1 Overview

We implemented the system Javara based on our reconstruction method. Javara is currently designed to run on a web browser. This choice of design may seem eccentric, but this method has an advantage. Implementation of Javara on a web browser enables editors to reduce the trouble of checking page appearance during their work.

4.2 Typical flow of restructuring task

To begin editing with Javara, an editor inputs a URI of the top page of the target site (see Figure 6). Javara CGI (Perl scripts) converts the page source to add functions for editing. Also, the CGI stores the original contents for backup. After converting the page source, the editor can start refining the site through the web browser, and immediately view responses to changes.

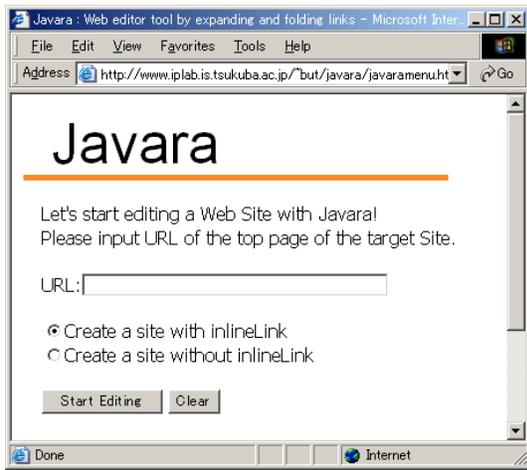


Figure 6: Start page of Javara editing

The editor can select the anchor types: inlineLink or a normal anchor. If the editor chooses the inlineLink option, the anchors of the top page are converted. Outputs of Javara are reconstructed HTML files. After editing the site, an editor selects the menu “Save the site.” The editor can obtain the output files from a location assigned by the system.

4.3 Components of the system

Javara consists of two components, a file manager component and an interface component. The file manager stays in the server, and the interface works from the client side.

The file manger component mainly treats inputs and outputs of the system. At the beginning of the editing, the file manger component converts the top page into an editable state and stores it as a temporary file. At the end of the editing, the system transfers the temporary file into the files needed to build the site as the result of editing.

The interface component covers the functions for editor operations for editing based on the proposed method. We implemented the dynamic editing functions with a Dynamic HTML mechanism.

4.4 Unfolding and folding features

We realized the unfolding and folding operations by modifying DOM objects and their properties. The modification of the objects immediately reflects the appearance of the contents.

First, the system replaces each link anchor in the target page with an “unfolding function anchor” that calls the function of unfolding. Also, the system wraps each anchor in a element with a unique ID to specify regions of insertion. If the anchor is unfolded, the

innerHTML of the corresponding SPAN element is replaced as the linked page source.

To fold a region, the system should specify what portion is to be exported in HTML source. However, in contrast to unfolding, an editor can fold any region desired, so that it is impossible to prepare . Therefore, the system should identify the region of the HTML source by matching with the text that the editor selected.

The system replaces the specified region with a pseudo unfolding function anchor. The content of the replaced region is stored as an argument of the pseudo unfolding function. When an editor clicks the pseudo unfolding anchor, the content is inserted into the base page.

4.5 Zooming feature

We realized zooming by dynamically modifying the style sheet of the document. In order to fit the page into the current window height, the system calculates a suitable zooming rate from the height of the document and the window. The system then detects the current focusing region. While performing context zooming, the system first calculates the focusing part, then allocates the available area to the rest of the part to fit the window.

4.6 Generating editable page contents

JavaScript functions and some extra objects are added to the top page to enable the features for the proposed editing method: unfolding, folding, zooming display, searching with keywords, cutting&pasting text and so on. The link anchors are replaced with the anchors to the JavaScript functions for unfolding. Finally, the transferred file is given to the interface system.

When other pages on the site are unfolded and involved in editing, the links are replaced and functions are added automatically. Replacing the anchor, the system checks the URI and link situation of each page and adds extra codes to warn the loop of links, and so on.

4.7 Generating restructured files

The interface component cannot write files because it runs as a client JavaScript. After editing, the interface component sends the contents of the current HTML source and the structure of the site to the file manager component that runs on the server side. The current HTML source that includes the site structure and its contents is copied to the VALUE property of an invisible FORM object. Consequently, the content information is passed via a POST method. The file manager analyzes the contents and creates the new pages and their

links from the argument of function declaration. After constructing the necessary files, the file manager deletes unnecessary Javara function codes in the contents.

5 Applicable scope of our method

Here we will discuss the web sites that can be edited with the Javara system. The Javara system modifies web site hierarchy structures to restructure a web site. However, web sites may not have freely linked hierarchy structures. The structure of such sites may be a complex graph. Such a structure cannot be edited by Javara alone, but can be edited by exception handling as we described in Section 3. Editors do not need to edit the whole site; they can simply edit a part of the site separately by starting to edit from the top page of the partial hierarchy structure in the site with Javara. Most web sites have hierarchy structures, so Javara editing can be applied.

6 Related work

Denim[5] is a system that helps web site designers in the early stages of the site design. Designers can construct the hierarchy of their web site by sketching both page items and relationships among them on a zoomable pane. The Denim technique is useful in prototyping of site designing but cannot reconstruct existing web pages. Also, the designer has to check each page with a web browser after editing. In the Javara system, the designer does not have to do this because Javara performs reorganization on a web browser.

Much researches has been conducted on improving the web usability by considering site structure, access logs and so on. ReWeb[6] analyzes both site structure and access logs. After classifying the site structure into typical ones, the viewer shows the structure. WebQuilt[7] shows web trace logs collected by a proxy as a visual graph. The viewer also employs a zoomable interface. The site designers can acquire a typical path of the site. These techniques help site designers to find problems of their site structure. After finding these problems, site developers should fix them. At that stage, we believe that our Javara system will serve as an effective tool for site reconstruction incorporating an overview.

7 Future work

The Javara system is designed for reduced-step operation, which means no confirmation in typical heavy

editing such as unfolding and folding effects. This approach is better for expert editors but is not convenient for beginners. Also, the system should support no-effect unfolding, which means not changing the page structure. The no-effect unfolding operation can buffer the site restructuring and verification tasks.

8 Conclusion

In this paper, we described “Javara,” a site-restructuring system based on link expanding and unfolding. Javara helps editors control content by introducing two concepts: an unfolding link and folding. An unfolding link inserts a linked page content by replacing the link with the link-following page. Folding exports the part of a document selected by an editor to a new page, then simultaneously creates a link to the page. Owing to these concepts, Javara reduces the steps of these tasks. These content rearrangement tasks are essential for restructuring web sites. Our Javara system can thus be useful, especially for hierarchical documents.

References

- [1] Pawan R. Vora. Design/Methods & Tools: Designing for the Web: A Survey. *interactions*, Vol. 5, Issue 3, pp. 13–30, May/June 1998.
- [2] Microsoft. Microsoft FrontPage. <http://www.microsoft.com/frontpage/>.
- [3] MacroMedia. MacroMedia DreamWeaver. <http://www.macromedia.com/software/dreamweaver/>.
- [4] Motoki Miura, Buntarou Shizuki, and Jiro Tanaka. inlineLink: Inline Expansion Link Methods in Hypertext Browsing. In *Proceedings of International Conference on Internet Computing (IC2001)*, pp. 653–659, June 2001.
- [5] James Lin, Mark W. Newman, Jason I. Hong, and James A. Landay. DENIM. In *Proceedings of the CHI 2000 conference on Human factors in computing systems*, pp. 510–517, April 2000.
- [6] Filippo Ricca and Paolo Tonella. Understanding and Restructuring Web Sites with ReWeb. *IEEE MultiMedia*, Vol. 8, No. 2, pp. 40–51, April-June 1998.
- [7] Jason I. Hong and James A. Landay. WebQuilt: A Framework for Capturing and Visualizing the Web Experience. In *Proceedings of the tenth World Wide Web conference on World Wide Web*, pp. 717–724, May 2001.